Freeform Search

Database:	US Pre-Grant Publication Full-Text Database US Patents Full-Text Database US OCR Full-Text Database EPO Abstracts Database JPO Abstracts Database Derwent World Patents Index IBM Technical Disclosure Bulletins
Term:	
Display:	10 **Documents in Display Format:** - **Starting with Number** 1
Generate:	○ **Hit List** ● **Hit Count** ○ **Side by Side** ○ **Image**

Search Clear Interrupt

Search History

DATE: Thursday, June 17, 2004 Printable Copy Create Case

Set Name side by side	**Query**	**Hit Count**	**Set Name** result set
	DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR		
L23	l6 and (photography or photographs or pictures)	150	L23
	DB=USPT; PLUR=YES; OP=OR		
L22	5359512.pn.	1	L22
L21	5465373.pn.	1	L21
	DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR		
L20	l2 and (task near2 manag$ or project near2 manag$)	228	L20
	DB=USPT; PLUR=YES; OP=OR		
L19	5029115.pn.	1	L19
L18	5113356.pn.	1	L18
L17	5581667.pn.	1	L17
L16	5581667.pn.	1	L16
L15	5625766.pn.	1	L15
L14	5666543.pn.	1	L14
L13	5666543.pn.	1	L13
L12	5713032.pn.	1	L12
L11	5761392.pn.	1	L11

L10	5988899.pn.	1	L10
L9	6011905.pn.	1	L9
L8	6012070.pn.	1	L8
L7	6049390.pn.	1	L7
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L6	L5 and copie$	206	L6
L5	L4 and low near2 resolution	473	L5
L4	L3 and high near2 resolution	1177	L4
L3	L2 and host	4100	L3
L2	L1 and digital near2 image	19530	L2
L1	(www or internet or network or web)	1430552	L1

END OF SEARCH HISTORY

First Hit Fwd Refs

□ Generate Collection Print

L6: Entry 171 of 206 File: USPT Dec 18, 2001

US-PAT-NO: 6332146
DOCUMENT-IDENTIFIER: US 6332146 B1

TITLE: Method and apparatus for storing and printing digital images

DATE-ISSUED: December 18, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Jebens; John H.	Tierra Verde	FL		
James; Jeffrey Scott	Bettendorf	IA		
Carlson; Lowell D.	Moline	IL		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Marshall, O'Toole, Gerstein, Murray & Borun	Chicago	IL			02

APPL-NO: 09/ 619188 [PALM]
DATE FILED: July 19, 2000

PARENT-CASE:
This is a Continuation of U.S. application Ser. No. 08/908,046, filed Aug. 11, 1997.

INT-CL: [07] G06 F 17/30

US-CL-ISSUED: 707/104; 707/3, 707/10, 707/102, 705/26, 705/27, 345/428
US-CL-CURRENT: 707/104.1; 345/428, 705/26, 705/27, 707/10, 707/102, 707/3

FIELD-OF-SEARCH: 707/3, 707/10, 707/102, 707/104, 707/530, 705/27, 705/33, 705/42, 705/26, 345/132, 345/302, 345/418, 345/428, 382/284, 382/276, 382/249, 386/124, 355/40, 355/70, 396/639

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL Clear

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
□	4495582	January 1985	Dessert et al.	700/124
□	4688181	August 1987	Cottrell et al.	345/428

□	4799156	January 1989	Shavit et al.	705/26
□	4933880	June 1990	Borgendale et al.	707/515
□	4956769	September 1990	Smith	707/9
□	4992940	February 1991	Dworkin	705/26
□	5065447	November 1991	Barnsley et al.	382/249
□	5144556	September 1992	Wang et al.	707/9
□	5153936	October 1992	Morris et al.	345/428
□	5263157	November 1993	Janis	707/9
□	5276901	January 1994	Howell et al.	707/9
□	5315693	May 1994	Hirosawa	345/428
□	5319401	June 1994	Hicks	354/76
□	5319543	June 1994	Wihhelm	705/3
□	5414811	May 1995	Parluski et al.	345/501
□	5426594	June 1995	Wright et al.	709/206
□	5440401	August 1995	Parulski et al.	386/124
□	5463555	October 1995	Ward et al.	700/96
□	5469353	November 1995	Pinsky et al.	382/131
□	5493677	February 1996	Balogh et al.	707/104
□	5539906	July 1996	Abraham et al.	707/9
□	5553281	September 1996	Brown et al.	707/104
□	5581749	December 1996	Hossain et al.	707/10
□	5584022	December 1996	Kikuchi et al.	707/9
□	5606365	February 1997	Maurinus et al.	348/222
□	5625776	April 1997	Johnson	705/27
□	5630125	May 1997	Zellweger	707/103
□	5666215	September 1997	Fredlund et al.	358/487
□	5696901	December 1997	Konrad	709/203
□	5784461	July 1998	Shaffer et al.	705/51
□	5845263	November 1999	Camaisa et al.	705/27
□	5852435	December 1998	Vigneaux et al.	345/302
□	5875268	February 1999	Miyake	382/276
□	5978804	November 1999	Dietzman	707/10
□	6017157	January 2000	Garfinkle et al.	396/639
□	6154755	November 2000	Dellert et al.	707/526

OTHER PUBLICATIONS

"Japan-Israel venture to offer photo development through Internet", Major Articles, Nikkel English News, Nov. 8, 1996.

"Konica Offers On-Line Photo Processing--Only in the US", Computer International, No. 2914, May 16, 1996.

"Photofinishing Comes to the Web", Newsbytes News Network, Mar. 7, 1996.

"The Internet", Israel Technology and Investment Letter, vol. 2, No. 2, Mar. 1, 1996.

Martin, James A., "Moving Images Without Tears," Macworld, v12, n12, p. 121 (2), Dec., 1995.

"Methods For Image Management," (Seybold Special Report, Part II), Seybold Report on Publishing Systems, v24, n18, p. S44 (6), May 15, 1995.

"More Notes From Nexpo: Image Handling, Digital Cameras and Links To Presses; Image Handling: Archiving, Retrieval, Etc.," (Includes A Related Article On Companies Offering Newspapers Pre-Made Comic Pages), Seybold Report on Publishing Systems, v25, n1, p. 28 (11), Sep. 1, 1995.

Richards, Kathleen, "PhotoNet addresses PC photos. (Internet-based service offers high-quality output of photographic material for personal computer users)", The Weekly Newspaper for the Home Furnishing Network, vol. 70, No. 51, Dec. 16, 1996.

Smith, Jeff, "Konica's Photo Service is Going On-Line by May", Business Tuesday, Pulse, Portland Press Herald, Mar. 12, 1996.

Straus et al., "Netw@tch The AJC'S Daily Online Guide Signing on for photos, customers can save trip to camera shop", Features, Atlanta Journal and Constitution, Feb. 20, 1996.

Symons, Allene, Beam me up a phot greeting card, Scotty. (drugstores and electronic on-line photos and cards from Konica and American Greetings), Drug Store News, vol. 18, No. 6, Apr. 1, 1996.

Wildstrom, Stephen H., "Technology & You: Bulletin Board: Snapshots . . . Or Via The Web", Business Week, No. 3497, Oct. 14, 1996.

"Supermarket Photo Service Touts Snapshots on the Net", Interactive Media Briefs, Interactive Marketing News, vol. 3, Issue 35, Dec. 6, 1996.

"PC PICS", Finance, Business Briefcase, Boston Herald, Dec. 4, 1996.

"Photos Over Net", Business, Local, Telegram & Gazette, Worcester, MA, Dec. 4, 1996.

"A CRW Report", News, Computer Retail Week, Dec. 2, 1996.

"Plaza Create to make electronic photo albums", Japan Computer Industry Scan, Nov. 18, 1996.

"Plaza Create enjoys first rise in four days", Tokyo and Osaka Stock Markets, Nikkei English News, Nov. 14, 1996.

"Plaza Create (7502) to sell color printer, digital camera", Major Articles, Nikkei English News, Nov. 13, 1996.

"Telecommunications & Technology: Plaza Create to Form Venture with U.S.", The Wall Street Journal Europe, Nov. 11, 1996.

Baig, Edward C., "Smile--You're on Candid Computer. Software, scanners, and color printers are making digital photography a snap", Business Week, No. 3500, Nov. 4, 1996.

Bounds, Wendy, "Big Photo Retailer to Offer Service on the Internet", Marketing & Media, The Wall Street Journal, Feb. 21, 1996.

Buckler, Grant, "Startup MGI Going After Emerging Photo Software Market", Newsbytes, Nov. 27, 1996.

Elson, Joel, Hannaford, Shaw's see how Internet fits photos. (Hannaford Brothers Co. of Maine; Shaw's Supermarkets Inc. of Massachusetts; Internet use for photo finishing), Supermarket News, vol. 46, No. 47, Nov. 18, 1996.

Lansky, Jerry, "Without APS, Photo Life Goes on Via Internet", Photographic Trade News, Aug. 1, 1996.

Levin et al., "Web Photo Finish; New online services for shutterbugs", vol. 15, No. 19, PC Magazine, Nov. 5, 1996.

Miller, Leslie, "Web posting as a photo processing option", Life, USA Today, Dec. 13, 1996.

O'Neill, Jerry, "Photofinishers Shoot the Curl in Cyberspace", Net Gains, Photographic Trade News, Sep. 1, 1996.

Armstrong, "For This Printer, Scanning's a Snap", Business Week, p. 16, Aug. 11, 1997.

Rowley, "Israeli Firm Puts Photos On-Line", Chicago Tribune, Aug. 6, 1997.
Spinner, "Going With The Flow", CFO, table of contents and pp. 53-57, Aug. 1997.
Capturing & Saving Digital Images.
Manual: Media Assest Management, GISTICS Incorporated, 1997.
Brochure: Media Bank, Digital Asset Management by Archetype.
Brochure: Media Bank, The Power of Digital Asset Management by Archetype.
Brochure: Media Assets 1.6 by Media Way.
Brochure: Job Manager, Information Management System for the Graphic Arts Industry by Meta Communications.
Brochure: Telescope, Client Server Media Management Database by North Plains Systems, Inc.
Brochure: Luminous Media Manager by Luminous Technology Systems, Inc.
Brochure: Luminous Media Manager Background by Luminous Technology Systems, Inc.
Brochure: Destiny, Focusing the Power of Your Digital Information by Centillion Digital Systems.
Brochure: DAX, File Transfer by Digital Art Exchange, Inc.
Brochure: DAX Database Access by Digital Art Exchange, Inc.
Brochure: DAX, Remote Proofing by Digital Art Exchange, Inc.
Brochure: DAX, Computer-to-Plate by Digital Art Exchange, Inc.
Brochure: DAX, Interactive Mark-up by Digital Art Exchange, Inc.
Brochure: Digital Art Exchange, The Connectivity Solution by Digital Art Exchange, Inc.
Brochure: Cascade MediaSphere W3 by Cascade Systems, Inc.
Brochure: Cascade DataFlow by Cascade Systems, Inc.
Brochure: Cascade Product Overview by Cascade Systems, Inc.
Brochure: Cumulus Media Management System 3., by Canto.
Brochure: Hynet Digital Library System, Version 1.5 by Hynet Technologies.
Brochure: Galerie, Media Asset Management by Dalim.
Brochure: Dalim, Twist by Dalim.
Brochure: Cascade MediaSphere by Cascade Systems, Inc.
Brochure: Luminous PrintersWeb by Luminous Technology Corporation.

ART-UNIT: 212

PRIMARY-EXAMINER: Alam; Hosain T.

ASSISTANT-EXAMINER: Colbert; Ella

ATTY-AGENT-FIRM: Marshall, Gerstein & Borun

ABSTRACT:

A digital data management and order delivery system is provided. The system includes a storage device for storing digital data and a searching engine for developing a subset of the digital data stored in the storage device in response to inputs received from a first user. The system is also provided with a job order developer responsive to inputs received from the first user for developing a job order which includes: a) at least one copy of the digital data contained in the subset and identified by the first user; and b) a file containing information developed by the first user outside the system. In addition, the system includes a router for electronically routing the job order compiled by the job order developer to a second user specified by the first user.

25 Claims, 27 Drawing figures

First Hit Fwd Refs

□ Generate Collection Print

L6: Entry 172 of 206 File: USPT Nov 20, 2001

US-PAT-NO: 6321231
DOCUMENT-IDENTIFIER: US 6321231 B1
**** See image for Certificate of Correction ****

TITLE: Data management and order delivery system

DATE-ISSUED: November 20, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Jebens; John H.	Tierra Verde	FL		
Carlson; Lowell D.	Moline	IL		
James; Jeffrey Scott	Bettendorf	IA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Marshall, O'Toole, Gerstein, Murray & Borun	Chicago	IL			02

APPL-NO: 08/ 908046 [PALM]
DATE FILED: August 11, 1997

INT-CL: [07] G06 F 17/30

US-CL-ISSUED: 707/104; 707/10, 707/102, 345/428, 355/40, 355/70, 396/639
US-CL-CURRENT: 707/104.1; 345/428, 355/40, 355/70, 396/639, 707/10, 707/102

FIELD-OF-SEARCH: 707/3, 707/102, 707/530, 707/104, 705/33, 705/42, 345/132, 345/302, 345/428, 382/284, 382/276, 382/249, 395/200.36, 395/200.33, 586/124, 364/468.02, 355/40, 355/70, 396/639

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL Clear

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
□	4495582	January 1985	Dessert et al.	700/124
□	4688181	August 1987	Cottrell et al.	364/521
□	4799156	January 1989	Shavit et al.	364/401
□	4933880	June 1990	Borgendale et al.	364/523

□	4956769	September 1990	Smith	364/200
□	4992940	February 1991	Dworkin	364/401
□	5065447	November 1991	Barnsley et al.	382/249
□	5144556	September 1992	Wang et al.	364/419
□	5153936	October 1992	Morris et al.	345/428
□	5263157	November 1993	Janis	395/600
□	5276901	January 1994	Howell et al.	395/800
□	5315693	May 1994	Hirosawa	395/128
□	5319401	June 1994	Hicks	354/76
□	5319402	June 1994	Hicks	354/76
□	5319543	June 1994	Wihhelm	364/401
□	5383027	January 1995	Harvey et al.	358/296
□	5426594	June 1995	Wright et al.	395/200.36
□	5440401	August 1995	Parulski et al.	386/124
□	5463555	October 1995	Ward et al.	364/468.02
□	5469353	November 1995	Pinsky et al.	364/413.01
□	5493677	February 1996	Balough et al.	707/104
□	5539906	July 1996	Abraham et al.	395/600
□	5553281	September 1996	Brown et al.	395/600
□	5581749	December 1996	Hossain et al.	395/600
□	5584022	December 1996	Kikuchi et al.	395/609
□	5625776	April 1997	Johnson	395/227
□	5630125	May 1997	Zellweger	395/614
□	5666215	September 1997	Fredlund et al.	358/487
□	5696901	December 1997	Konrad	395/200.33
□	5784461	July 1998	Shaffer et al.	386/21
□	5845263	November 1999	Camaisa et al.	705/27
□	5852435	December 1998	Vigneaux et al.	345/302
□	5875268	February 1999	Miyake	382/276
□	5978804	November 1999	Dietzman	707/10
□	6017157	January 2000	Garfinkle et al.	396/639
□	6154755	November 2000	Delbert et al.	707/526

OTHER PUBLICATIONS

Armstrong, "For This Printer, Scanning's a Snap", Business Week, p. 16, Aug. 11, 1997.
Rowley, "Israeli Firm Puts Photos On-line", Chicago Tribune, Aug. 6, 1997.
Spinner, "Going With The Flow", CFO, table of contents and pages 53-57, Aug. 1997.
Capturing & Saving Digital Images.

Manual: Media Asset Management, GISTICS Incorporated, 1997.
Brochure: Media Bank, Digital Asset Management by Archetype.
Brochure: Media Bank, The Power of Digital Asset Management by Archetype.
Brochure: Media Assets 1.6 by Media Way.
Brochure: Job Manager, Information Management System for the Graphic Arts Industry by Meta Communications.
Brochure: Telescope, Client Server Media Management Database by North Plains Systems, Inc.
Brochure: Luminous Media Manager by Luminous Technology Systems, Inc.
Brochure: Luminous Media Manager Background by Luminous Techology Systems, Inc.
Brochure: Destiny, Focusing the Power of Your Digital Information by Centillion Digital Systems.
Brochure: DAX, File Transfer by Digital Art Exchange, Inc.
Brochure: DAX Database Access by Digital Art Exchange, Inc.
Brochure: DAX, Remote Proofing by Digital Art Exchange, Inc.
Brochure: DAX. Computer-to-Plate by Digital Art Exchange, Inc.
Brochure: DAX, Interactive Mark-up by Digital Art Exchange, Inc.
Brochure: Digital Art Exchange, The Connectivity Solution by Digital Art Exchange, Inc.
Brochure: Cascade MediaSphere W3 by Cascade Systems, Inc.
Brochure: Cascade DataFlow by Cascade Systems, Inc.
Brochure: Cascade Product Overview by Cascade Systems, Inc.
Brochure: Cumulus Media Management System 3., by Canto.
Brochure: Hynet Digital Library System, Version 1.5 by Hynet Technologies.
Brochure: Galerie, Media Asset Management by Dalim.
Brochure: DALIM, Twist by Dalim.
Brochure: Cascade MediaSphere by Cascade Systems, Inc.
Brochure: Luminous PrintersWeb by Luminous Technology Corporation.
Martin, James A., "Moving Images Without Tears," Macworld, v12, n12, p. 121 (2), Dec., 1995.
"Methods For Image Management," (Seybold Special Report, Part II), Seybold Report on Publishing Systems, v24, n18, p. S44 (6), May 15, 1995.
"More Notes From Nexpo: Image Handling, Digital Cameras and Links To Presses: Image Handling: Archiving, Retrieval, Etc.," (Includes A Related Article On Companies Offering Newspapers Pre-Made Comic Pages), Seybold Report on Publishing Systems, v25, n1, p. 28 (II), Sep. 1, 1995.

ART-UNIT: 212

PRIMARY-EXAMINER: Alam; Hosain T.

ASSISTANT-EXAMINER: Colbert; Ella

ATTY-AGENT-FIRM: Marshall, O'Toole, Gerstein, Murray & Borun

ABSTRACT:

A digital data management and order delivery system is provided. The system includes a storage device for storing digital data and a searching engine for developing a subset of the digital data stored in the storage device in response to inputs received from a first user. The system is also provided with a job order developer responsive to inputs received from the first user for developing a job order which includes: a) at least one copy of the digital data contained in the subset and identified by the first user; and b) a file containing information developed by the first user outside the system. In addition, the system includes a router for electronically routing the job order compiled by the job order developer to a second user specified by the first user.

74 Claims, 27 Drawing figures

First Hit Fwd Refs

□ Generate Collection Print

```
L6: Entry 190 of 206                    File: USPT                Mar 16, 1999
```

US-PAT-NO: RE36145
DOCUMENT-IDENTIFIER: US RE36145 E

TITLE: System for managing tiled images using multiple resolutions

DATE-ISSUED: March 16, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
DeAguiar; John R.	Sebastopol	CA		
Larkin; Ross M.	Rolling Hills	CA		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
Optigraphics Corporation	San Diego	CA			02

APPL-NO: 08/ 559027 [PALM]
DATE FILED: November 16, 1995

REISSUE-DATA:

US-PAT-NO	DATE-ISSUED	APPL-NO	DATE-FILED
05263136	November 16, 1993	694416	April 30, 1991

INT-CL: [06] G06 F 13/00

US-CL-ISSUED: 345/511; 345/501, 395/200.3, 382/232
US-CL-CURRENT: 345/538; 345/501, 345/555, 382/232, 709/200

FIELD-OF-SEARCH: 395/128, 395/139, 395/114, 395/501, 395/502, 395/507, 395/508, 395/511, 395/200.01, 395/200.02, 395/200.3, 395/200.31, 345/189, 345/201, 345/190, 345/202, 345/428, 345/439, 345/501, 345/502, 345/507, 345/508, 345/511, 345/509, 382/232, 382/244, 382/248, 382/240, 382/238

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL Clear

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
□	Re31200	April 1983	Sukonick et al.	395/162
□	4878183	October 1989	Ewart	395/128
□	4920504	April 1990	Sawada et al.	395/166

□	4951230	August 1990	Dalrymple et al.	395/166
□	4969204	November 1990	Melnychuck et al.	382/240
□	5020003	May 1991	Moshenberg	395/164
□	5138459	August 1992	Roberts et al.	348/232
□	5150462	September 1992	Takeda et al.	395/166
□	5568570	October 1996	Rabbani	382/238

OTHER PUBLICATIONS

"Addresss Generation and memory management for memory centered image processing systems" by Reader et al, pp. 88-96, Procedings of SPIE-The International Society for Optical Engineering, V757, Methods of Handling and Processing Imagery, Jan. 15-Jan 16, 1987.

ART-UNIT: 273

PRIMARY-EXAMINER: Tung; Kee M.

ATTY-AGENT-FIRM: Knobbe, Martens, Olson & Bear, LLP

ABSTRACT:

An image memory management system for tiled images. The system defines an address space for a virtual memory that includes an image data cache and a disk. An image stack for each source image is stored as a full resolution image and a set of lower-resolution subimages. Each tile of an image may exist in one or more of five different states as follows: uncompressed and resident in the image data cache, compressed and resident in the image data cache, uncompressed and resident on disk, compressed and resident on disk and not loaded but re-creatable using data from higher-resolution image tiles.

39 Claims, 46 Drawing figures

First Hit Fwd Refs

□ Generate Collection Print

L6: Entry 190 of 206 File: USPT Mar 16, 1999

DOCUMENT-IDENTIFIER: US RE36145 E
TITLE: System for managing tiled images using multiple resolutions

Brief Summary Text (5):
The present invention relates to memory management systems and, more particularly, to the memory management of large digital images.

Brief Summary Text (7):
The present invention comprises a memory management system for large digital images. These digital, or raster, images are made up of a matrix of individually addressable pixels, which are ultimately represented inside of a computer as bit-maps. Large digital images, such as those associated with engineering drawings, topographic maps, satellite images, and the like, are often manipulated by a computer for the purpose of viewing or editing by a user. The size of, such images are often on the order of tens and even hundreds of Megabytes. Given the current cost of semiconductor memory it is economically impracticable to dedicate a random access memory (RAM) to storing even a single large digital image (hereinafter just referred to as a "digital image"). Thus, the image is usually stored on a slower, secondary storage medium such as a magnetic disk, and only the sections being used are copied into main memory (also called RAM memory).

Brief Summary Text (9):
Presently, to enhance main memory storage of only relevant sections of a digital image, the image is logically segmented into rectangular regions called "tiles". Two currently preferred standards for segmenting an image into tiles are promulgated by the Computer Aided Logistics Support (CALS) organization of the United States government (termed the "CALS standard" herein) and by Aldus Corporation of Seattle, Washington, as defined in the Tagged Image Format File (TIFF) definition (e.g., "TIFF Specification, Revision 5.0, Appendix L). Among other tile sizes, both standards define a square tile having dimensions of 512.times.512 pixels. Thus, if each pixel requires one byte of storage, the storage of one such tile would require a minimum of 256 kilobytes of memory.

Brief Summary Text (12):
Even when stored in a mass storage system, an image library, containing a number of digital images, will consume disk space very quickly. Furthermore, "raw" digital images are generally too large to transfer from mass storage to portable floppy disks, or between computer systems (by telephone, for example), in a timely and inexpensive manner unless some means is used to reduce the size of the image. Hence, users of binary images employ image compression techniques to improve storage and transfer efficiencies. One existing compression standard applicable to facsimile transmission, CCITT Group IV, or T.6 compression, is now being used for digital images. Like many other compression techniques, however, the CCITT standard uses statistical techniques to compress data and, hence, it does not always produce a compressed image that is smaller than the original, uncompressed image. That means that image libraries will often contain a mix of compressed and uncompressed binary images. Similar compression standards exist for color and gray-scale images such as those promulgated by the JPEG (Joint Photog. Exp. Group) Standards Committee of the CCITT as SGV III Draft Standard.

Brief Summary Text (13):
At the present time, digital images are typically viewed and modified with an image editor using an off-the-shelf computer workstation. These workstations usually come with a sophisticated operating system, such as UNIX, that employs a virtual memory to effectively manage memory accesses in secondary and main memories. In an operating system having virtual memory, the data that represents the executable instructions for a program or the variables used by that program do not need to reside entirely in main memory. Instead, the operating system brings portions of the program into main memory only as needed. (The data that is not stored in main memory being stored on magnetic disk or other like nonvolatile memory.) The address space that is available to any one application program is generally managed in blocks of convenient sizes called "pages" or "segments".

Brief Summary Text (17):
However, the management of virtual memory for images departs significantly from conventional virtual memory schemes because images and computer programs are accessed in very different ways. Computer programs tend to access one small neighborhood of virtual address, and then jump to some distant, essentially random, location. However, during normal image processing operations an image is accessed in one of a finite set of predictable patterns. It is not surprising then that conventional memory management systems can significantly degrade performance when used in image processing applications by applying inappropriate memory management rules. Rules which should be abided by a memory management system for large digital images are the following:

Brief Summary Text (22):
5. An image memory management system should support the capability to "undo" editing operations which is built into the memory manager for optimal performance and ease of use. Thus, the memory manager could easily save copies of the compressed tiles in the affected region, and quickly restore the image to the original state by simply modifying the tile directory entries to point to the old version.

Drawing Description Text (35):
FIG. 34 is a flow diagram defining the "create compressed low resolution tile from compressed higher-resolution tiles" function referred to in FIG. 31;

Drawing Description Text (36):
FIG. 35 is a flow diagram defining the "copy uncompressed high resolution tile to uncompressed low resolution tiles" function referred to in FIG. 31;

Detailed Description Text (4):
In order to compensate for lower performance expected with a virtual memory management system for images, particularly when reducing large portions (by combining pixels) of the image for display, the present invention automatically maintains a series of reduced resolution copies, called subimages, of the full resolution image. Preferably, the resolution (i.e., pixels per inch) of each subimage is reduced by exactly half relative to the next higher-resolution subimage. Thus, the image stack 100 can be visualizing as an inverted pyramid, wherein the images can be stacked beginning with a full resolution subimage (or image) 102 at the top, followed by a half resolution subimage 104, then a quarter resolution subimage 106, and an eighth resolution subimage 108. (In FIG. 1, the subimages 102-108 are outlined by bolded lines.)

Detailed Description Text (12):
The I/O bus 168 also supports the hard disk drive 162, and the Ethernet communications port 164. A hard disk controller 176 connects the hard disk drive 162 to the I/O bus 168. The hard disk drive 162, in one possible configuration of the workstation generally indicated at 150, stores 60 megabytes of data. An Ethernet communications controller 178 connects an Ethernet communications port 164

with the I/O bus 168. The Ethernet communications controller 178 supports the industry standard communications protocol TCP/IP which includes FTP and Telnet functions. The Ethernet communications port 164 of the preferred embodiment allows the Workstation 150 to be connected to a network which may include, among other things, a document scanner (not shown) and a print server (not shown).

Detailed Description Text (19):
To use the image data cache 194, the memory management functions begin by determining how much fast memory (RAM) and slow memory (disk or host memory) is available for image memory uses. When an image is loaded, the system allocates memory for image information and related tile directory structures. Cache management parameters are modified as necessary to balance the requirements for expanded tile and compressed tile cache memory. The expanded tile cache memory pool and the compressed tile cache memory pool allow tiles from different images to intermingle. Expanded and compressed tiles are kept in separate areas as much as possible so that memory allocation can be optimized for each of two different situations (i.e., fixed allocation block size versus variable size). However, the storage ranges of compressed and expanded tiles are allowed to mingle so as to maximize the flexibility of the cache usage.

Detailed Description Text (22):
The eight transformation operations, shown in square boxes in FIG. 8, constitute the main computational operations associated with managing image memory. The operation "load compressed tile image data from disk into cache memory" 264 is typically the first operation performed on a tile because most pre-scanned images are stored in compressed form in disk files. (A discussion of this "virtual loading" is provided hereinbelow.) The load operation 264 is performed by the Load CompFromDisk function which simply copies data from the disk into cache memory. The disk location and number of bytes to read is stored in the tile header fields 368 and 376 shown in FIG. 10.

Detailed Description Text (34):
302 Self-reference to document handle. Handle value assigned to this document by the host procedure which created the document. This value is unique over the entire system.

Detailed Description Text (87):
The function LoadSubImTile 436 calls a function Create Compressed Lower-Resolution Tile From Compressed Higher-Resolution Tiles 442 (CompCopyToOview). CompCopyToOview creates a valid compressed version of the specified tile by scaling down from compressed or expanded version of the given higher-resolution subimage tiles. The function LoadSubImTile 436 also calls a function Copy Uncompressed High-Resolution Tiles to Uncompressed Low-Resolution Tile 444 (CopyTileToOview). CopyTileToOview updates the region of the next lower-resolution overview corresponding to the specified tile.

Detailed Description Text (118):
If an orthogonal rotation is specified then the tile manager 192, allocates a conventional access context at a state 542. Then the tile manager 192 continues to a decision state 544 wherein the subimage selection criterion is specified. For instance, the user may request the "low resolution" option which selects the lowest resolution subimage in the document's image stack. (In the context of an image editor, this may be the best solution during zooming or panning.) The user may also specify "most available"--i.e., whatever subimage has tiles currently in cache memory, regardless of the resolution. In either case, the tile manager 192 proceeds to a state 546 to select the reduced resolution subimage that is appropriate to that particular choice, i.e., either the one that has the resolution just greater than what was requested or a subimage whose tiles covering the access region are currently in cache. Now, at a state 548, the tile manager 192 adjusts the transformation matrix so as to now refer to the reduced resolution subimage rather

than the full resolution subimage by adjusting scale factors.

Detailed Description Text (123):
Then the tile manager 192 comes to a state 562, where the tile manager 192 allocates buffers for scaling, if necessary. This is the situation where intermediate copies of the rows or columns of data may need to be kept during the process of scaling. Then the tile manager 192 tests whether the user specified that the region needed to be saved for undoing, at a decision state 564.

Detailed Description Text (125):
Now at a decision state 568, the question is whether to update the subimages during the operation. If this is a write operation the tile manager 192 always writes into the full resolution subimage and the changes "trickle down" into the low resolution subimages. But the tile manager 192 has an option as to whether the lower-resolution tiles are updated during the modification operation or later when the tiles are requested for viewing operations. There are advantages in doing them both ways.

Detailed Description Text (126):
For example, if the affected region is small, it is more efficient to update the subimages while progressing through the operation. In this mode, when the tile is unlocked, the manager 192 immediately copies the data down into the next lower subimage tile but only one of the corners of the tile is affected. Thus, only portions of the low resolution subimage tiles need to be modified.

Detailed Description Text (171):
The tile manager 192 then moves to a state 694 where the tile manager 192 allocates memory for "undo region tile headers". These tile headers will be used to store copies of the original versions of the tiles in the affected region. The tile manager 192 then proceeds to a state 696 wherein the tile manager 192 makes an "undo tile directory".

Detailed Description Text (176):
If a valid copy of compressed data does reside on disk, the tile manager 192 moves to a state 714 and simply copies the compressed tile disk location and size information from the document tile header to the undo tile header. Note that it is possible for a particular tile to have multiple representations of the same data. That is, a compressed version and an expanded version of the tile may exist in cache simultaneously. And a tile may have a compressed version in cache as well as on the disk. For undo, the strategy is to store the most compact version possible. The most compact version with regard to cache memory usage is to have a copy of the compressed tile on the disk.

Detailed Description Text (177):
If there is no compressed copy of the tile on the disk, the tile manager 192 proceeds to a decision state 716 wherein the tile manager 192 determines whether an uncompressed copy of the document tile resides on the disk. If the test succeeds, the tile manager 192 enters a state 718 and copies the uncompressed tile disk location and size information from the document tile to the undo tile and then returns to the inner FOR-loop at a loop state 700.

Detailed Description Text (180):
If there was no error in locking cache memory at the state 730, the tile manager 192 moves to a state 732 and copies the compressed data from the document tile to the undo tile. The tile manager 192 actually copies the data that is stored within the tile--i.e., the compressed image data is copied from the document version to the undo version. Then the tile manager 192 moves to a state 734 and unlocks the compressed version of the document tile. Now, at a state 736, the tile manager 192 unlocks the compressed version of the undo tile and the tile manager 192 returns to the inner FOR-loop at state 700 on FIG. 17A where the tile manager 192 loops back

to continue the loop for all of the tiles in the affected region.

Detailed Description Text (188):
lr is the particular low resolution subimage resolution in pixels per inch.

Detailed Description Text (210):
From state 884, 882, or 880, as appropriate, tile manager 192 proceeds to a decision state 886 to check for polygonal clipping. If the tile manager 192 is doing polygonal clipping then each input row of data is clipped as appropriate for that polygon in states 888 and 890. The loop allows multiple clipped regions within each row. If there is no clipping, then the tile manager 192 simply copies the entire input row from the image into the input row buffer in a state 892. Then the tile manager 192 move to a state 894 where the tile manager 192 passes these input rows through the scaler if the tile manager 192 is scaling the data. Finally, the tile manager 192 takes the results of the scalers and copies that information to the output strip buffer if necessary at a state 896. The tile manager 192 then returns to the state 870 (shown in FIG. 22A) where the tile manager 192 continues the process of retrieving input rows and scaling them until the tile manager 192 has filled the output strip buffer. The system then moves to the termination condition at the end state 856.

Detailed Description Text (212):
Next, the tile manager 192 moves to state 916 where the tile manager 192 checks to see if the destination row index is outside of the image's clipping boundaries. If so, the tile manager 192 simply ignores it and moves back to state 914. If it is within the clip boundaries the tile manager 192 moves to state 918 where the tile manager 192 determines whether the destination row is in the currently locked tile row. If it is not, the tile manager 192 moves to state 920 where the tile manager 192 unpreserves and unlocks the old tile row that is currently locked. The tile manager 192 then moves to state 922 to determine whether the update overview flag is true. This is an option that is specified in the lo access context and it determines how lower-resolution tiles are updated when the full resolution subimage is modified. If the update overview flag is true, then the tile manager 192 moves to state 924 where the tile manager 192 unpreserves the low resolution tiles that will no longer be needed.

Detailed Description Text (213):
After the system has unpreserved the low resolution tiles that are no longer needed at state 924, the system moves to state 926 and locks down the new tile row. Only the full resolution tile row is locked at this level. The low resolution tiles are actually updated when the call to unlock the old tile row is made.

Detailed Description Text (215):
The tile manager 192 then moves to state 934, wherein the tile manager 192 conducts a FOR-loop for each of the clip point pairs that the tile manager 192 computed in state 930. As shown in FIG. 23B, the tile manager 192 loops to state 936 where the tile manager 192 copies pixels from a scaler output buffer to the image row between each pair of clip points. When that loop terminates, the tile manager 192 returns to state 914 in FIG. 22A. On the other hand, if the tile manager determines at state 930 that polygonal clipping is not active, the tile manager 192 moves to state 938, wherein the tile manager 192 copies the scaler output buffer pixels to the image row without clipping. The tile manager 192 then proceeds to state 914.

Detailed Description Text (224):
At state 996, shown in FIG. 25B, the tile manager 192 determines whether the undo tile is marked as blank. If it is, the tile manager 192 moves to state 998, wherein the tile manager marks the document tile as blank, and then the system loops back to state 986. If the undo tile is determined to be not blank at state 996, the tile manager 192 move to state 1000. At state 1000, the tile manager 192 checks to see if the undo tile points to compressed data on the disk. If it does, the tile

manager 192 moves to state 1002 and copies the disk location and size information about the compressed data into the document tile header and loops back around. If there is no compressed data on the disk, then the tile manager 192 moves from state 1000 to state 1004, wherein the tile manager 192 determines whether uncompressed data exists on the disk associated with the undo tile.

Detailed Description Text (225):
If so,the tile manager 192 moves to state 1006, wherein the file manager 192 copies the disk location and size information about the uncompressed data into the document tile header and loops back to state 986. If the system determines at state 1004 that there is no uncompressed data on the disk, the tile manager 192 proceeds to state 1008, wherein the tile manager 192 determines whether the undo tile "points" to uncompressed data in cache memory. If it does, the tile manager 192 moves to state 1010, wherein the tile manager 192 copies the pointer to the uncompressed data from the undo header to the document tile header.

Detailed Description Text (248):
If, at the state 1112, the tile is found to be locked, the tile manager 192 moves to a state 1116 to decrement the lock count. Thereafter, the execution continues to a decision state 1118 wherein the tile manager 192 tests whether the "update overview" flag is set true. If the flag is set, the tile manager 192 moves to a state 1120 to update the corresponding lower-resolution tiles. In the process of modifying tiles, the tile manager 192 locks a tile down in the image data cache to write to it. When the tile is unlocked, that is a signal to the memory manager to update the lower resolution tiles that correspond to the higher resolution tile. Thus, the image data in the high resolution tile being unlocked is copied down into the lower resolution tiles, all the way down to the bottom of the image stack.

Detailed Description Text (256):
Assuming that all the higher-resolution tiles have been loaded, the FOR-loop terminates and the tile manager 192 proceeds to test whether all of the higher-resolution tiles are blank. If all four of the high resolution tiles mapped to this low resolution are blank, the tile manager 192 transitions to a state 1160 to mark the low resolution tile as blank. The tile manager 192 does not create any image data for the blank, lower-resolution tile. The tile manager 192 and terminates the function 436 at the end state 1144.

Detailed Description Text (258):
Now, if it is determined that it is faster to scale the expanded version of the data, the tile manager 192 moves from the state 1162 to a state 1166 to allocate memory for the uncompressed version of the lower-resolution tile. From the state 1166, the tile manager 192 moves to the beginning of a FOR-loop at a loop state 1168 wherein for each of the higher-resolution tiles the tile manager 192 scales the expanded version of the higher-resolution tile directly into the proper position in the lower-resolution tile using the function 444. When the tile manager 192 has scaled each of the four high resolution tiles, the tile manager 192 has completed the creation of the expanded version of the low resolution tile.

Detailed Description Text (271):
FIG. 34 illustrates a process for creating compressed low resolution tiles from compressed higher resolution tiles. The tile manager 192 starts at start state 1250 and proceeds to state 1252, wherein the system enters a loop which is followed by the system for each of the four high resolution tiles required to produce a single low resolution tile. More specifically, at state 1252 the tile manager 192 locks the compressed version of the high resolution tile. The system then proceeds to state 1256, wherein the tile manager 192 determines whether an error occurred at state 1254. In the event that an error occurred, the tile manager proceeds to end state 1258 and terminates. If no error occurred, the tile manager 192 returns to state 1252 and continues the loop described above for each of the four high resolution tiles.

Detailed Description Text (272):
After processing all four high resolution tiles as described, the system proceeds to state 1260 where the tile manager 192 scales the compressed data to half resolution. The process performed at state 1260 results in a compressed version of the low resolution tile. Then the tile manager 192 moves to a loop represented by states 1262, 1264, wherein for each of the high resolution tiles the tile manage 192 unlocks the compressed version of the tile.

Detailed Description Text (273):
Next, the tile manager 192 moves to state 1266 where the tile manager 192 allocates and locks memory for the compressed version of the low resolution tile. At state 1266, the tile manager 192 actually puts the compressed version of the low resolution tile in a general, common buffer that is large enough to hold the maximum possible size of the compressed results. The actual valid data is usually much less than that than the maximum possible size, so the tile manager 192 only saves the valid amount of data.

Detailed Description Text (274):
From state 1266, the system moves to state 1268 to determine whether an error occurred at state 1266. If an error occurred, the system moves to end state 1258 and terminates. Otherwise, the system moves to state 1270 where the tile manager 192 copies the compressed data out of the temporary compressed data buffer into the newly allocated space in the cache. Then the tile manager 192 moves to state 1272 where the tile manager 192 unlocks the compressed version of the low resolution tile that now contains valid data. The system then terminates normally at state 1258.

Detailed Description Text (275):
Now referring to FIG. 35, a process is shown whereby the system resamples uncompressed high resolution tiles to an uncompressed low resolution tile. The tile manager 192 starts at start state 1280 and moves to state 1282, wherein the tile manager 192 locks the uncompressed version of a single high resolution tile. This function scales a single high resolution tile to update one quarter of a tile in the half-resolution subimage. That quarter tile is rescaled to update one-sixteenth of a tile in the quarter-resolution subimage. This continues to the lowest resolution subimage. Next, the tile manager 192 proceeds to state 1284 to determine whether an error occurred in locking the uncompressed version of the high resolution tile. If there Was an error, then the tile manager 192 proceeds to state 1286 and terminates with an error condition. Otherwise, the tile manager 192 moves to state 1288 where the tile manager 192 determines how many levels of the subimage are to be updated. This function can be used to update a subset of subimages or the entire image stack in the case where a single tile is modified in the full resolution subimage. It will propagate that change all the way down to the lowest-resolution subimage in the image stack.

Detailed Description Text (276):
Next, the tile manager 192 proceeds to state 1290 where the tile manage 192 determines the tile index that is to be updated. In accordance with the present invention, when a change is propagated from the higher resolution down to the low resolution of tiles, the system calculates which tile corresponds to the affected area. Then the tile manager 192 moves to state 1290 where the tile manager 192 determines whether the low resolution tile that the tile manager 192 is about to update is marked as loaded or not. This step is intended for the situation in which not all of the low resolution substates are populated during the loading of a raster image.

Detailed Description Text (277):
If the system determines that one or more low resolution tiles are not loaded, the system proceeds to state 1294, wherein the tile manager 192 invalidates all of the

low resolution tiles that would otherwise be affected by the change. The system then exits normally at end state 1286. If the low resolution tile is about to be modified is loaded, as determined at state 1292, the tile manager 192 moves to state 1296, wherein the system locks the uncompressed version of the low resolution tile. The tile manager 192 then moves to state 1298 to determine whether an error occurred at state 1296 and, if so, the system moves to end state 1286 to terminate. Otherwise, the system moves to state 1300. wherein the tile manager 192 scales the raster data from the high resolution tile down to the low resolution tile. Then the tile manager 192 moves to state 1302 where the tile manager 192 unlocks the high resolution tile.

Detailed Description Text (278):
Next, the system moves to state 1304, wherein the tile manager 192 recursively modifies the loop variables such that the low resolution tiles that the tile manager 192 just finished updating become the high resolution tiles for the next succeeding iteration. Once all the subimages have been updated as described, the system exits at end state 1286.

Detailed Description Text (285):
Next, the tile manager 192 moves to state 1344, wherein the system determines whether the request made at state 1342 has been satisfied. If so, the system terminates at end state 1346. Otherwise, the system moves to state 1348, wherein the tile manager 192 frees unlocked, but preserved uncompressed tiles that have valid compressed or uncompressed copies. The tile manager 192 preferentially frees the oldest such tiles.

Detailed Description Text (296):
From state 1420, the system proceeds to state 1422 to determine whether an error occurred at state 1420. If an error occurred, the system moves to state 1418 and proceeds as described above. Otherwise, the system moves to state 1424, wherein the tile manager 192 copies the compressed data from the common buffer into the newly allocated cache memory state. The system moves from state 1424 to state 1426, wherein the tile manager 192 unlocks the compressed and uncompressed tile data and then terminates at end state 1408.

CLAIMS:

1. An image memory management system, comprising:

a computer having a processor and an image memory, the image memory comprising a main memory and a secondary memory;

an image stack, located in the image memory, comprising a plurality of similar digital images, each digital image having a plurality of pixels grouped into at least one tile, and each digital image having a resolution different from the other digital images;

means for accessing a selected one of the tiles in the image stack;

first means for transferring a selected one of the tiles from the secondary memory to the main memory when the tile is accessed by the accessing means and the tile is absent from the main memory; and

second means for transferring a selected one of the tiles from the main memory to the secondary memory when the main memory is full.

7. The system defined in claim 1, wherein a lowest resolution digital image comprises one tile.

8. The system defined in claim 1, wherein a preselected digital image in the image

stack is resampled to obtain another digital image in the image stack.

9. The system defined in claim 1, wherein at least one of the digital images is compressed.

13. A method of managing images in a computer .Iadd.system .Iaddend.having a processor and an image memory comprising a slower access memory and a faster access memory, comprising the steps of:

creating a digital image;

resampling the digital image so as to form an image stack comprising the digital image and one or more lower resolution digital images;

dividing each image into equal sized, rectangular tiles; and

evaluating a location in the image memory of tiles in each digital image of the image stack in a given region of interest.

17. The method defined in claim 13, wherein the evaluating step includes selecting the digital image with the lowest resolution higher than a requested resolution al a given view scale. .Iadd.

18. The system defined in claim 1, wherein the image memory is located in a network..Iaddend..Iadd.19. The system defined in claim 1, wherein the image memory is located in the Internet..Iaddend..Iadd.20. The system defined in claim 9, wherein the form of compression is Joint Photographic Experts Group (JPEG)..Iaddend..Iadd.21. The method defined in claim 13, wherein at least one of the digital images is compressed..Iaddend..Iadd.22. The method defined in claim 21, wherein the form of compression is Joint Photographic Experts Group (JPEG)..Iaddend..Iadd.23. The method defined in claim 13, wherein the computer system comprises a plurality of computers connected together in a network..Iaddend..Iadd.24. The method defined in claim 23, wherein the network comprises the Internet..Iaddend..Iadd.25. An image memory management system, comprising:

a computer system having a processor, a main memory and a secondary memory;

an image stack, comprising plural digital images, each digital image divided into at least one region, at least two of said digital images having different resolutions;

a selector which selects one of said regions;

a list identifying regions in said main memory;

a sensor determining when said main memory is full;

a data exchanger responsive to said selector, said list and said sensor to move said one region from the secondary memory to the main memory when said region is selected by said selector and said list indicates that the region is absent from the main memory, and to move regions from the main memory to the secondary memory when said sensor indicates that the main memory is full..Iaddend..Iadd.26. The system defined in claim 25, wherein at least one of the digital images is compressed..Iaddend..Iadd.27. The system defined in claim 26, wherein the form of compression is Joint Photographic Experts Group (JPEG)..Iaddend..Iadd.28. The method defined in claim 25, wherein the computer system comprises a plurality of computers connected together in a network..Iaddend..Iadd.29. The method defined in claim 28, wherein the network comprises the Internet..Iaddend..Iadd.30. A method of managing images in a computer system comprising the steps of:

resampling a digital image to form an image stack comprising plural digital images of differing resolutions, wherein each digital image is displayable independent of the other images of the image stack;

selecting an image resolution; and

selecting an image from said image stack for display, based on said selected image resolution..Iaddend..Iadd.31. The method defined in claim 30, wherein at least one of the digital images is compressed..Iaddend..Iadd.32. The method defined in claim 31, wherein the form of compression is Joint Photographic Experts Group (JPEG)..Iaddend..Iadd.33. The method defined in claim 30, wherein the computer system comprises a plurality of computers connected together in a network..Iaddend..Iadd.34. The method defined in claim 33, wherein the network comprises the Internet..Iaddend..Iadd.35. A method of managing images in a computer system comprising the steps of:

accessing a digital image;

resampling the digital image so as to form an image stack comprising the digital image and one or more lower resolution digital images, wherein each digital image is displayable independent of the other images of the image stack; and

selecting a portion of one of the digital images in the image stack for

display..Iaddend..Iadd.36. The method defined in claim 35, wherein at least one of the digital images in the image stack is compressed..Iaddend..Iadd.37. The method defined in claim 36, wherein the form of compression is Joint Photographic Experts Group (JPEG)..Iaddend..Iadd.38. The method defined in claim 35, wherein the computer system comprises a plurality of computers connected together in a network..Iaddend..Iadd.39. The method defined in claim 38, wherein the network comprises the Internet..Iaddend.

First Hit Fwd Refs

□ Generate Collection Print

L6: Entry 195 of 206 File: USPT Feb 24, 1998

DOCUMENT-IDENTIFIER: US 5721835 A
**** See image for Certificate of Correction ****
TITLE: Information processing system, electronic device and control method

Detailed Description Text (9):
The following units are connected to the PCI bus: a VRAM 105 for developing display image data; a liquid-crystal display device (LCD) 106 for displaying the image developed in the VRAM 105; and a graphic controller 107 which executes processing for painting images in the VRAM 105, extracts image from the VRAM 106 and delivers it to the LCD 106 as a video signal. It should be noted that the graphic controller 107 is also equipped with a D/A converter for converting digital bit image data, which has been developed in the VRAM 105, to an analog signal. This output signal is supplied to an extension interface, described later. Also connected to the PCI bus are a power management unit (POWER MGMT) 108, which manages the power supplied from a chargeable battery 126 and, when the capability of the supply becomes too small, causes a message to this effect to be informed the CPU 101, and an IDE controller 109, which controls writing and reading of an IDE-type hard disk 110.

Detailed Description Text (17):
In the illustration, a LAN card 212 is loaded in interface 207 and a graphic controller card 213 is loaded in the interface 208. Though no explanation is necessary, the LAN card 212 is an extension card (board) for constructing a local area network. The graphic controller card 213 mounts a VRAM of several megabytes. Though the number of display dots of the LCD in the notebook PC 100 is, say, 640.times.400 dots, the graphic controller 213 extends this to 1024.times.768 dots. In addition, the graphic controller 213 executes processing for developing bitmap images, raises the speed of movement and increases the number of colors produced.

Detailed Description Text (18):
Though the details will be described later, the graphic controller card 213 merely delivers the input video signal (a CRT signal) to the CRT 300 as is without operating in order to display the image (low resolution) from the VRAM 105 of the notebook PC 100. In this case, the CPU 101 applies various paint instructions to the graphic controller 107 within the notebook PC 100 and causes the display image to be developed in the VRAM 105 of the notebook PC 100.

Detailed Description Text (19):
In a case where a high-resolution, high-speed display using the capability of the graphic controller card 213 is to be presented, the CPU 101 applies various draw and paint instructions to the graphic controller card 213. At such time the graphic controller card 213 ignores the CRT signal sent from the notebook PC 100 and outputs the image data stored in its own VRAM to the CRT 300 as the video signal. If the graphic controller card has not been installed, the video signal from the notebook PC 100 is delivered to the CRT as is.

Detailed Description Text (36):
In a case where it is determined that the connected notebook PC has not been registered, the program proceeds to step S13, at which registration processing (described later) is executed. In order for the registration procedure to be carried out, at the very least the keyboard 400 with which the dock 200 is provided

is enabled. To accomplish this, ID=0 is temporarily assigned to the notebook PC 100. It should be noted that the content of the VRAM 105 of the notebook PC 100 continues to be displayed at low resolution on the CRT display unit 300.

Detailed Description Text (46):
As a result of the foregoing, it becomes possible for only a limited number of users to use the confidential files in a host or server by employing resources, such as the hard disk connected to the dock, as well as a network card in some cases.

Detailed Description Text (55):
In a case where a different OS is selected by selecting "2" (the OS that has been stored on the hard disk of the dock) or "3" (an OS stored in a server on a network) in the OS selection menu of FIG. 12, the environment is decided and the selected OS is started up in accordance with the devices allowed for use in the resources of the notebook PC and dock.

Detailed Description Text (60):
If it is judged that the OS is incapable of running, then a message is displayed to the effect that the designated OS cannot be used. In addition, the program returns to step S43 in order that another OS may be selected. For example, if, in a case where booting from a network has been designated, the user has been forbidden from using the network card 212, a message to this effect is displayed.

Detailed Description Text (103):
The foregoing example describes the relationship between the hard disk 214 in the dock 200 and the main memory 102 and hard disk 110 in the notebook PC 100. Operation is the same in a case where a file in a relevant directory of the hard disk 214 in dock 200 is copied or moved to another directory (route directory) and in a case where the reverse is carried out.

Detailed Description Text (107):
Thus, when the user of the notebook PC 100 works with an application program after the notebook PC has been connected to the dock 200, writes a file under an encryption directory of the hard disk 214 or executes save processing, the data is automatically subjected to encryption processing in accordance with the ID of the notebook PC connected. Further, in a case where a file in the encryption directory is read in, the file is not read in as normal data so long as the notebook PC connected to the dock is not that of an authorized user; hence, file confidentiality can be maintained. In addition, a convenient feature is that merely moving the file from the directory to another location makes it possible for the encrypted data to be read freely and viewed by another user. In this case the fact that a file can viewed by another user means that the file can be read in using the application that created it and can be edited. Further, since a file server on a network can be treated as a virtual storage device, byway of example, it can be readily surmised that it will be possible to cope with this situation as well.

Detailed Description Text (116):
In accordance with the description given above, when the operator moves or copies a file located within his or her own encryption directory to a directory for which encryption has not been designated by this operator, the encryption file is converted to a data file that is capable of being decoded. If it so happens that the directory that is the destination of the moved or copied file has been designated as an encryption directory by the user of another notebook PC, then even though it is not required to perform decoding when this user reads in this file, erroneous decoding processing is executed in accordance with the ID of the user's own notebook PC.

Detailed Description Text (117):
In a case where a file is copied without notice from another party's encryption

directory to one's own encryption directory, a message file created in one's own encryption directory may be transferred to the other party's encryption directory (particularly another user's encryption directory located within another dock connected to a network). Though this is convenient from the viewpoint of maintaining security, a problem that arises is the other party will be unable to decode the message.

Detailed Description Text (119):
(1) In a case where the directory that is the destination of the transfer is the encryption directory of another user, the transform is forbidden and a message to this effect is displayed for the operator to read. Though this makes it possible to preclude the aforesaid problem, this technique is not especially suited to an information exchange via a network. Although no problem is encountered in a case where a shared non-encryption directory is created and a file is copied to this directory, an unspecified number of individuals can read in the file and view its contents. Hence, intrusion by a third party cannot be prevented.

Detailed Description Text (130):
Specifically, there are cases in which the notebook PC 100 is connected to the dock 200, the CRT display unit 300 is used in the high-resolution mode, processing proceeds with the window of an application being displayed along one edge of the display screen (the right edge or lower edge if the upper left-hand corner of the window is taken as the origin) and then processing is terminated. If the notebook PC is subsequently detached from the dock and the same application is started up using the LCD display of the notebook PC, the window will come to be displayed in virtual space off the screen of the LCD. If this happens, the user will not be able to use the application or, even if the application can be used, it will be necessary for the user to move and resize the window and make it fit inside the screen.

Detailed Description Text (160):
Further, use can be limited user by user at the discretion of the supervisor. For example, access to a network can be denied to a novice operator, thereby making it possible to preclude accidents such as the inadvertent deletion of an important file. When the operator has become sufficiently skilled, the scope of permission given to the operator can be broadened.

First Hit Fwd Refs

□ Generate Collection Print

```
  L23: Entry 137 of 150                    File: USPT              Mar 28, 2000

US-PAT-NO: 6043909
DOCUMENT-IDENTIFIER: US 6043909 A

TITLE: System for distributing and controlling color reproduction at multiple sites

DATE-ISSUED: March 28, 2000

INVENTOR-INFORMATION:
 NAME                      CITY          STATE     ZIP CODE       COUNTRY
 Holub; Richard A.         Brighton      NY

ASSIGNEE-INFORMATION:
 NAME                     CITY          STATE  ZIP CODE  COUNTRY   TYPE CODE
 Imagicolor Corporation   Rochester     NY                          02

APPL-NO: 08/ 606883   [PALM]
DATE FILED: February 26, 1996

INT-CL: [07] H04 N 1/23

US-CL-ISSUED: 358/504; 358/518
US-CL-CURRENT: 358/504; 358/518

FIELD-OF-SEARCH: 358/504-505, 358/518-523, 358/400, 358/500, 382/162, 382/167

PRIOR-ART-DISCLOSED:

                           U.S. PATENT DOCUMENTS
```

Search Selected Search ALL Clear

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
□	2790844	April 1957	Neugebauer	178/5.2
□	4500919	February 1985	Schreiber	358/78
□	4987496	January 1991	Greivenkamp, Jr.	358/448
□	5107332	April 1992	Chan	358/80
□	5109275	April 1992	Naka et al.	358/80
□	5182721	January 1993	Kipphan et al.	364/526
□	5185673	February 1993	Sobol	358/296
□	5200816	April 1993	Rose	358/80
	5231481	July 1993	Eouzan et al.	358/60

□				
□	5272518	December 1993	Vincent	356/405
□	5309257	May 1994	Bonino et al.	358/504
□	5319437	June 1994	Van Aken et al.	356/326
□	5345315	September 1994	Shalit	358/406
□	5394204	February 1995	Shigeta et al.	353/31
□	5432906	July 1995	Newman et al.	395/162
□	5465111	November 1995	Fukushima et al.	347/115

OTHER PUBLICATIONS

Heidelberger Druckmashinen Aktiengesellschaft, "In-Line Image Control CPC 23", pp. 1-4.
C. Soderg.ang.rd, T. Lehtonen, R. Launonen, & Juuso Aikas, "A System For Inspecting Colour Printing Quality", Reprinted from TAGA Proceedings, vol. 1, The Technical Association Of The Graphic Arts, Rochester, New York, pp. 620-634, 1995.
R. Blessing, "Before Your Very Eyes", Publishing & Production Executive, Aug. 1995.

"Graphic Technology-Color Reflection Target For Input Scanner Calibration", American National Standard, Annex B, pp. 18-20, Jun. 1993. (ANSI.RTM.IT, Jul. 2-1993).
H. Boll, "A Color To Colorant Transformation For A Seven Ink Process", SPIE, vol. 2170, pp. 108-118, 1994.
"International Color Consortium Profile Format", Version 3.01, May 8, 1995.
C. Sodergard, M. Kuusisto, Y. Xiaohan, K. Sandstrom, "On-line Control of the Colour Print Quality Guided by the Digital Page Description", Reprinted from Proceeding of 22nd International Conference On Printing Research Institutes, Munich, Germany, 1993.
J. Gordon & R. Holub, "On the Use of Linear Transformations for Scanner Calibration", Communications and Comments, Color Research & Application, vol. 18, No. 13, pp. 218-219, Jun. 1993.
"MCT.TM. Metric Color Tag Specification-Draft", Revision 1.1d, EfiColor Developers Desk, Electronics for Imaging, Inc., pp. 1-30, Mar. 22, 1993.
R. H. Johnson, & D. W. Wichern, Applied Multivariate Statistical Analysis, 3rd Edition, Prentice Hall, Chapter 8, pp. 356-395, 1992.
A. Hardy & F. Wurzburg, Jr., "Color Correction In Color Printing", Journal Of The Optical Society Of America, vol. 38, No. 1, pp. 300-307, Apr. 1948.
H. E. J. Neugebauer, "The Colorimetric Effect Of The Selection Of Printing Inks And Photographic Filters On the Quality Of Multicolor Reproductions", TAGA Proceedings, Technical Association Of The Graphic Arts, Rochester, New York, pp. 15-29, 1956.
I. Pobboravsky, "A Proposed Engineering Approach To Color Reproduction", TAGA Proceedings, Technical Association Of The Graphic Arts, Fourteen Annual Meeting, Rochester, New York, pp. 127-165, Jun. 11-13, 1962.
R. H. Gallagher, Finite Element Analysis: Fundamentals, Pentice-Hall, Englewood Cliffs, New Jersey, Chapter 8, pp. 228-241, 1975.
W. K. Pratt, Digital Image Processing, New York: Wiley, Chapter 19, pp. 551-559, 1978.
Commission Internationale de L'Eclairage, "Colorimetry", Second Edition, Austria, Publication CIE 15.2, pp. 19-23, 27-32, 1986.
J. Gordon, R. Holub & R. Poe, "On the Rendition of Unprintable Colors", TAGA Proceedings, Technical Association Of The Graphic Arts, Rochester, New York, pp. 1-10, 1987.
R. Holub, W. Kearsley & C. Pearson, "Color Systems Calibration for Graphic Arts: I. Input Devices", Journal Of Imaging Technology vol. 14, No. 2, pp. 47-52, Apr. 1988.

R. Holub, W. Kearsley & C. Pearson, "Color Systems Calibration For Graphic Art: II. Output Devices", Journal Of Imaging Technology, vol. 14, No. 2, pp. 53-60, Apr. 1988.
R. Holub & W. Kearsley, "Color to colorant conversions in a colorimetric separation system", SPIE, vol. 1184, Japan, pp. 24-35, Dec. 1989.
J. Greivenkamp, "Color dependent optical prefilter for the suppression of aliasing artifacts", Applied Optics, vol. 29, No. 5, pp. 676-684, Feb. 10, 1990.
C. Sodergaard, I. Ylakoski, & H. Vanhala, "A General Teleproofing System", Reprinted from In: Proceedings of the TAGA Conference, Rochester, New York, pp. 88-99, May 1991.
"TIFF.TM.", Revision 6.0 Final--Jun. 3, 1992, Aldus Corporation, Seattle, Washington, pp. 13-16, 1992.
C. Hoyt, "Toward Higher Res, Lower Cost Quality Color and Multispectral Imaging", Reprinted from Advanced Imaging, Apr. 1995.
"Abstracts of Awards for Fiscal Year 1995", SBIRP, U.S. Department of Commerce, p. 46.
"Experience The Future", Komori World News 28, International Edition, p. 5, Sep. 1995.
"Introducing Imagexpo.TM. 1.2 Interactive remote viewing and annotation software for the graphic arts professional", Group Logic, Arlington, Virginia, 1994-1995.
W. Press, B. Flannery, S. Teukolsky, & W. Vetterling, "Numerical Recipes The Art of Scientific Computing", Cambridge University Press, Chapter 14.3, pp. 509-520, Chapter 5.3, pp. 137-140, Chapter 9.6, pp. 269-273, & Chapter 10.8-10.9, pp. 312-334, 1986.
R. Holub, "The Future Of Parallel, Aanlog And Neural Computing Architectures In The Graphic Art", TAGA Proceedings, pp. 80-112, 1988.
Y. Garini, Ph.D., "Thin-Film Measurements Using SpectraCube.TM., Application Note For Thin Film Measurements", Spectral Diagnostics (SD) Ltd., pp. 1-6, Jan. 1995.
M. Schwartz, R. Holub, & J. Gilbert, "Measurements of Gray Component Reductiion in Neutrals and Saturated Colors", TAGA Proceedings, pp. 16-27, 1985.
R. Holub, "Colorimetric Aspects Of Image Capture", Track III-Image Processing, IS&T's 48th Annual Conference Proceedings, Arlington, Virginia, pp. 449-451, May 1995.
Ocean Optics, Inc., Price and Data Sheet for Spectrometers, Mar. 1, 1995.

ART-UNIT: 274

PRIMARY-EXAMINER: Moore; David K.

ASSISTANT-EXAMINER: Brinich; Stephen

ATTY-AGENT-FIRM: LuKacher, Esq.; M. LuKacher, Esq.; K. J.

ABSTRACT:

The system provides for controlling color reproduction of input color image data representing one or more pages or page constituents in a network having nodes (or sites). Each one of the nodes comprises at least one rendering device. The system distributes the input color image data from one of the nodes to other nodes, and provides a data structure (virtual proof) in the network. This data structure has components shared by the nodes and other components present only at each node. Next, the system has means for providing color calibration data at each node characterizing output colors (colorants) of the rendering device of the node, and means for producing at each node, responsive to the color calibration data of the rendering device of the node, information for transforming the input color image data into output color image data at the rendering device of the node. The information is then stored in the data structure in different ones of the shared and other components. Means are provided in the system for transforming at each node the input color image data into output color image data for the rendering

device of the node responsive to the information in the data structure. The rendering device of each node renders a color reproduction of the page constituents responsive to the output color image data, wherein colors displayed in the reproduction at the rendering device of each node appear substantially the same within the output colors attainable by the rendering devices. The system further has means for verifying at each node that the information for the rendering device of the node properly transformed the input color image data into the output color image data, and means for revising the information stored in the data structure at the node responsive to results of the verifying means. Shared components of the data structure may also store color preferences selected by a user. The information producing means of the system may further operate responsive to both the color calibration data and the color preferences. The rendering devices in the system can provide color reproductions having three or four colorants, and may provide more than four output colors (color inks).

69 Claims, 41 Drawing figures